



Project Remedies Inc.

ITIL v 3: What the CIO and CFO Need to Know

*Transaction Costs and the Concept of Tasking
Or... How to Save \$625,000 Per Year, Every Year,
in Your IT Organization.*

A White Paper from Project Remedies Inc.

February 26, 2009



Abstract

The Information Technology Infrastructure Library (ITIL) is a set of concepts and policies for managing the entire IT organization: infrastructure, development, and operations. ITIL is published in a series of books, the most recent is ITIL v3, which was published in May 2007. The rapid adoption of ITIL methodologies is a hot topic because it gives IT management clearly defined “best practices,” consistent processes, on how to set up and measure IT services.

The first topic in the first ITIL v3 book, Service Strategy, is transaction costs. The authors explain why you need to know the cost of each transaction which comes into your IT organization. Each transaction is called a “service request”. Service requests and their associated costs are increasing across the enterprise to meet the needs of the business.

When demand for services exceeds what can be delivered, senior IT and business managers are seeking help to prioritize and choose the IT services that deliver the greatest business value. The only way you can make sound business decisions that will allow you to survive or even thrive in today's economy is to base these decisions on accurate, timely, and complete cost information. Having consistent processes and a seamless, easy way to measure their costs, becomes the issue.

Determining Costs

One of the difficulties in determining your costs is that the programs used to track different types of work and to collect costs are fragmented and seldom located within the same system. IT shops normally look for the “best” product for each type of service request they perform. There are 3 types of service requests (at a high level), and 3 types of costs: time (resource costs), expenses, and asset costs. If all of this information is stored in disparate systems, you will need to do what someone called the “\$3 - \$5 million never-ending integration project” to bring it all together so you can do basic cost analysis.

Another problem with storing information across disparate products is the difficulty it causes for doing Resource Management. Resource Management answers at least the following questions:

- “What is everyone working on?”
- “Do we have enough resources to do what we want to do?” This is sometimes called “aligning resources to the business’s priorities.”



- “When, where and in which organizations will we have skills surpluses and skills shortages?”

One of characteristics of managing resources is that the list of tasks people are working on changes all the time. The list is very fluid. In order to better direct them to work on the highest priority tasks, you need to have one list.

We know what people are working on: the work that makes up their jobs. Work is represented by tasks: incidents and problems, changes, project tasks (which are 3 types of service requests), and other tasks not related to requests such as meetings or managing the servers. But if the resource task data is stored in 2, 3, 4, or even 5 or more different applications, the reality is that you will never be able to answer basic resource questions because you won't be able to see all the information gathered together in one place. This is what one CIO called “controlling runaway IT costs”.

BMC Remedy Action Request System

One of the products used in many IT organizations is the BMC Remedy Action Request System, which is a development environment/workflow engine. BMC has developed a suite of applications called the IT Service Management suite, which is ITIL certified and runs on the Action Request System environment. This means that the business processes defined in the ITSM applications work the way ITIL says a system should work. This application suite manages service requests, incidents and problems, changes, and assets. The asset management module keeps track of asset costs.

Project Remedies' ActionPortfolio Manager and ActionProgram Manager run on your existing Action Request System platform. If you add them to the BMC Remedy IT Service Management suite, the combined Service Management product suite handles all 3 types of service requests and all 3 types of costs – *all within the same system*. This is done without the need for a major, costly, and time-consuming integration project.

This combination also allows you to manage resources most effectively and enhances reporting capabilities.

When taking into consideration the cost of a major and time consuming integration project, PRI's software applications cost a fraction of what other portfolio and program management solutions cost. They are also much easier to integrate, and much easier to use. PRI's applications leverage your investment in Remedy in multiple ways.



Contents

Situation Overview

Tracking Requests and Their Costs, and the Concept of Tasking

- Example Scenario I - A Developer's Workday

- Example Scenario II - A Field Technician's Workday

Types of Requests

Types of Costs

The Problem

The Project Remedies' Solution

One More Thing



Situation Overview

Managing transaction costs is the very first topic discussed in the first section of the Introduction to Service Strategy, the first book in the ITIL v3 Series (page 3). The authors explain:

“Sometimes it makes sense for a business to own and operate assets, or conduct activities in-house. At other times, the sensible thing is to seek alternatives from the open market. As prevailing conditions change, boundaries of the firm contract or expand with decisions as *make, buy, or rent*. (Authors italics.)”

The decision to “make, buy or rent” involves many factors, one being cost. Being the low-cost provider is just one consideration. But how do you know if you are the low-cost provider, or if one of the outsourcers you are considering is the low-cost provider? The only way to know is to know what your costs are. You have read about “running IT like a business.” This is the same topic: this is business. *In order to run your business, you must know what your costs are.*

If an outsourcer you are considering proposes that they will charge \$X per request to fulfill a certain type of request, the only way to know if that is a good price is to know how much it currently costs you to fulfill that type of request.

Tracking Requests and their Costs, and the Concept of Tasking.

In an IT organization, work can be divided into two types: tasks that are performed as a result of a user request, and tasks that are not performed as a result of a user request. The word “task” is used to represent any element of work. Your IT employees usually perform both types of tasks during the normal workday.

Example Scenario I – A Developer's Workday

A developer begins his day working on an assigned task for a new application: writing code and testing a particular feature. The task he is working on is part of a project plan. He is interrupted to provide help to a user who has a question about an application the developer worked on previously. That task assignment (called an “incident”) comes from the help desk. The developer is also supposed to be working on a bug that began as an incident from the help desk, but turned out to be a bigger problem. In the afternoon, the developer attends an HR meeting where a new company policy is explained.

The new application the developer is working on is the result of an extended proposed/active project process that involves multiple layers of management. It began with a request from the VP of Sales for new functionality. Eventually



in the process, a detailed project work plan was created, and our developer was assigned responsibility for some of the tasks in the plan.

Answering the Help Desk call to help the user might take the developer only a few minutes, but it might take longer if research and/or testing are necessary to determine the answer.

Fixing a bug might involve just that bug, but it might just as easily uncover several related bugs that are then grouped together as a small project. Regardless, fixing the problem is part of a change management process, because the fix has to be implemented, typically by someone in another group.

On this day, our developer worked on 3 tasks generated from user requests and spent time at a meeting that was not part of a project plan or user request but just participating as an employee of the company.

To understand the **actual cost** of each request, the time spent by the developer on the tasks related to each request must be tracked. Who initiated the request and the process it took to get to the person working the task varies greatly depending on the type – and typically the urgency – of the request.

With this approach, all work, i.e., every task associated with every request, is tracked. Obviously, the key is having an easy way to create and track tasks, and that is exactly what PRI's applications provide.

Example Scenario II – A Field Technician's Workday

Another example scenario is that of the field technician who installs workstations and other types of hardware. This person also went to the same afternoon HR meeting as the developer, to understand the new company policy. The field tech spent most of the rest of the day implementing hardware that came in response to different user requests.

Our field technician has been assigned the following task: “Install a workstation on Bob Smith’s desk.” This task could potentially have come from 3 separate service requests, each with a different process life-cycle:

1. Bob called the help desk and explained that he needs a workstation.
2. HR submitted their standard request for a new employee named Bob Smith, and one of the many tasks is to provide him with a workstation.
3. The company President decided that all 5,000 people in the company need a new workstation, including one for Bob.

As we can see, making a service request is simple, but each type of request follows a different life-cycle and that's what differentiates one type of request from another: the life-cycle it follows. Two things happen in all of these examples: someone makes a request and a worker – a field technician or a



developer or someone with another job function – is eventually assigned a task to be performed. The differences between service request types have to do with the different processes or life-cycle each one follows. To understand the cost of each request, you need to know the cost of each task involved with fulfilling that request. Time is one element of that cost, and expenses and asset cost are the other elements. Time is more of a variable than expense and asset cost, because each person can do the job differently. (I do not want to get into a discussion about “Quality” here but understanding how much time is spent by different people performing the same task is critical to improving Quality.)

Types of Service Requests.

As we discussed above, at a high level, there are basically 3 types of requests and if you consider urgency, a 4th type. Each request type varies with the process or life-cycle it follows. A life-cycle normally involves an approval process and a work process. Within each type of request, there can be many different approval processes and work processes. But let’s focus on the high-level first.

These 4 types of requests might be called Type 1, Type 2, Type 3 and Type 4:

Type 1 is a small request. Only one task needs to be performed to satisfy the request. Someone on the service desk can approve the request. Bob calling and saying that his workstation is broken is a good example. A user calling for an answer to a question about an application is another good example.

Type 2 is a medium-sized request. After the request is approved, usually by a manager, a gaggle of tasks need to be performed, usually by different people and frequently in different organizations, to satisfy the request. You can have many different approval processes and many different work processes within this category.

Type 3 is a large request. It has more size and more dollar value. Because of this, it has a more extensive life-cycle. This life-cycle might start with a local approval process, then a more thorough “proposed project stage-gate” approval process which ends with the Management Steering Committee deciding that this is a good idea and requesting that a detailed project task plan be generated for their review. They want the detailed project task plan because they want a better estimate of the cost involved and the time involved with completing the project, i.e., when the benefits will be delivered, and will it cost what we think it will cost.

Once the project starts, each of the tasks is assigned to specific people to be performed, and the task plan is managed. The actual time incurred is compared with the baseline plan, and the duration of the task is compared with the baseline plan to make sure that the project stays within budget and



finishes on time. All projects, whether Development projects or Operations projects, go through this life-cycle.

Type 4 is a large request with so much urgency that the “proposed project stage-gate” approval process is skipped. A detailed project plan is developed, reviewed and approved. Tasks are assigned to different people, worked, and their time is entered and tracked by the project manager.

To summarize, no matter the service type, each process starts with a request and ends with a task being assigned to someone to work. In each case, people work the task, status the task, and enter their time against the task. What are different in each type are the steps between the request and the tasks assigned to people to be worked. Since we want to keep track of the time each task takes to complete, wouldn't it be great to have one system that handled all of the request types, their related tasks, one list of the people who work these tasks, and the assets that they work on?

Types of Service Costs.

Since we want to know the cost of each request, we need to track the 3 possible elements of cost for each request. The 3 cost elements are:

1. Time spent.
2. Expenses
3. Assets cost.

If we know the time spent on each request, we can multiply that number by a rate, usually an average rate, to come up with a cost. Rarely does a company actually use a person's salary in this calculation. Expenses incurred and the cost of the asset are added to the value of the time to come up with a total cost. Sometimes the asset cost is allocated to multiple requests or spread out over time. Time is also the biggest variable. It usually varies with the person or team performing the work.

If we know that it costs us \$50 on average to perform all of the tasks involved with setting up a new employee, if an outsourcer says that they will do it for \$100 per new employee, we know that we can do it in-house significantly less expensively than the outsourcer. If however, the outsourcer says that they will do it for \$51 per new employee, our internal cost is not that big of a difference, and cost will be less of a factor in making the decision to outsource or not. Cost is always a key factor but not the only factor.

The Problem

The problem is that IT management usually buys a different product to handle each type of request. If you want to capture time spent on tasks in different



systems, you then need to bring all of this data together, and this involves what someone accurately called the “\$3M - \$5M never-ending integration project.” Usually, these disparate products work on separate databases and platforms, but even if they all work on the same platform, the big problem is data integrity. This means that the rules in each product are different, so you are not bringing together apples and apples.

Recently, PRI successfully completed a consulting project for a large company that the client labeled a “consolidation and simplification project.” When our project leader asked what that means, the client said that they had 7 different help desk applications all built in Remedy, and the definition of a “new” request was different in all of them. If this happened in one company, what is the chance the definitions in different products created by different vendors are the same? The answer of course is “slim and none.” A movie studio spent \$7,000,000 with a major consulting firm to integrate their CA’s Clarity and BMC’s Remedy IT Service Management suite applications. A paper company in the Seattle area had a quote of \$2.7M to integrate MS Project Server and BMC’s Remedy’s IT Service Management suite.

Another problem with disparate applications is the inefficient use of enterprise resources. It takes them much more time to figure out their to-do list for the day if the data is in different systems. Also, if management wants to know what the people are working on, they have to look in 4 different systems, and you’ll never figure it out. Resource Management, i.e. knowing what everyone is working on, is vital to the efficient use of resources, which is a major factor in the company’s success.

The Project Remedies’ Solution

A better and more efficient approach, one that is less expensive, easier and faster to implement, is adding Project Remedies’ ActionPortfolio Manager and ActionProgram Manager to your existing BMC Remedy environment. BMC’s IT Service Management suite, includes approximately 2/3rds of the solution. It handles service requests, incidents and problems, changes, and asset management, i.e. the functionality to handle Type 1 and Type 2 requests, and the functionality to manage assets including asset costs. Project Remedies’ applications do the rest. Project Remedies’ applications can integrate with your existing request system to handle project initiation, has the functionality to manage type 3 and type 4 requests, and to do time and expense tracking against all Remedy-based tasks, as well as the time spent not working on tasks from user requests.

What immediate and unique features does this solution give you?

- It leverages your investment in your Remedy solution.



- It brings all costs (time, expenses, and assets) into one system.
- It is the only suite that manages projects and assets on the same environment.
- It eliminates the need for the \$3M - \$5M never-ending integration project.
- It allows management to look in one system to see what everyone is working on.
- It allows each user to look in one system to see what they are working on. If 300 people save 10 minutes a day by looking in one system rather than more, at \$50 per hour, the savings is \$625,000 per year. Every year. Do the math.
- It allows a manager to work with a user to better prioritize his work tasks, to better focus on what management's priorities.
- It allows users to enter time at the task level for all types of requests with the same system.
- It gives you the opportunity to easily define processes that cross application and organizational boundaries, improving efficiency and quality. Bug reporting is a good example. It eliminates the need to get a separate bug reporting system.
- It reduces the number of vendors that you have to deal with.
- It reduces the cost of user training because users are only dealing with one system rather than many.
- It reduces the cost of application support because your support team only supports one system, the BMC Remedy Action Request System.
- It allows you to incorporate program management functionality into your workflow processes.
- It allows you to report back to the head of each business unit exactly what you are doing for them at any time of the month. It tells the head of each business what his / her people are asking for, how much time you are spending working on these requests, and the cost incurred.
- It allows one system to be used for all of IT.



Finally, when demand for services exceeds what can be delivered, PRI Solutions helps you prioritize and choose the IT services that deliver the greatest business value.

One More Thing

A frustrated CFO recently stated, that IT management comes to him for his approval on the applications they want to buy or build using the resulting business benefits of the application for cost justification. He said that the differentiating feature(s) and benefits that make this the product of choice is usually not implemented until Phase 3 – and they never get to Phase 3. At Phase I, all of the products are about the same. With that in mind, he said that the PRI approach offered a terrific value and many important benefits – starting in Phase I.

All project management systems have project records and task records. They allow you to define complex task dependencies and usually inter-project dependencies. They all calculate start and finish dates using a Critical Path Method data calculator. They calculate slack and generate Gantt charts. They all do.

However, unlike traditional project management systems that only plan, schedule and report on project status, our solution allows your project managers to use task-level workflows to leverage ITIL v3 best-practice processes. Our applications are designed to be used for 9 different types of program management applications and can easily integrate with BMC's IT Service Management suite. PRI's applications leverage your investment in Remedy.

For more information about how the PRI solution could save \$625,000 per year, every year, in your IT organization, please call Stan Feinstein at 310-230-1722.